

Can We Do Better with What We Have Done? Unveiling the Potential of ML Pipeline in Notebooks



Bill (Yuangan) Zou
MAsc student



Xinpeng Shan
CS Undergrad



Shiqi Tan
MEng Student



Shurui Zhou
Assistant Professor

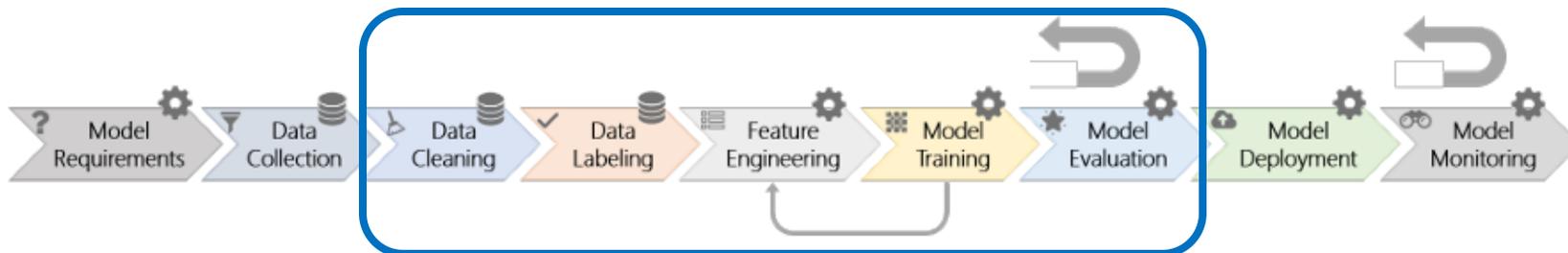


UNIVERSITY OF
TORONTO



Exploratory Programming in Notebooks

To derive insights from a large amount of data by building high-performance ML model.



Azure Notebooks



Exploratory Programming in Notebooks



- Linear structure
- Flexible
- Incremental

```
In [1]: import matplotlib.pyplot as plt
        from sklearn.cluster import KMeans
        from sklearn import datasets
```

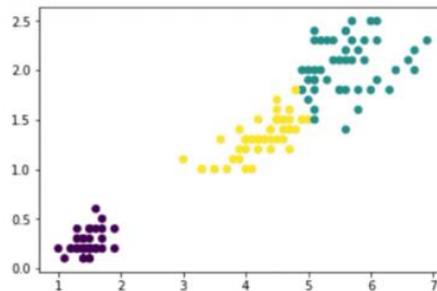
```
In [2]: data = datasets.load_iris().data[:,2:4]
        petal_length, petal_width = data[:,0], data[:,1]
```

```
In [3]: print("Average petal length: %.3f" % (sum(petal_length) / len(petal_length),))
Average petal length: 3.758
```

```
In [4]: clusters = KMeans(n_clusters=3).fit(data).labels_
```

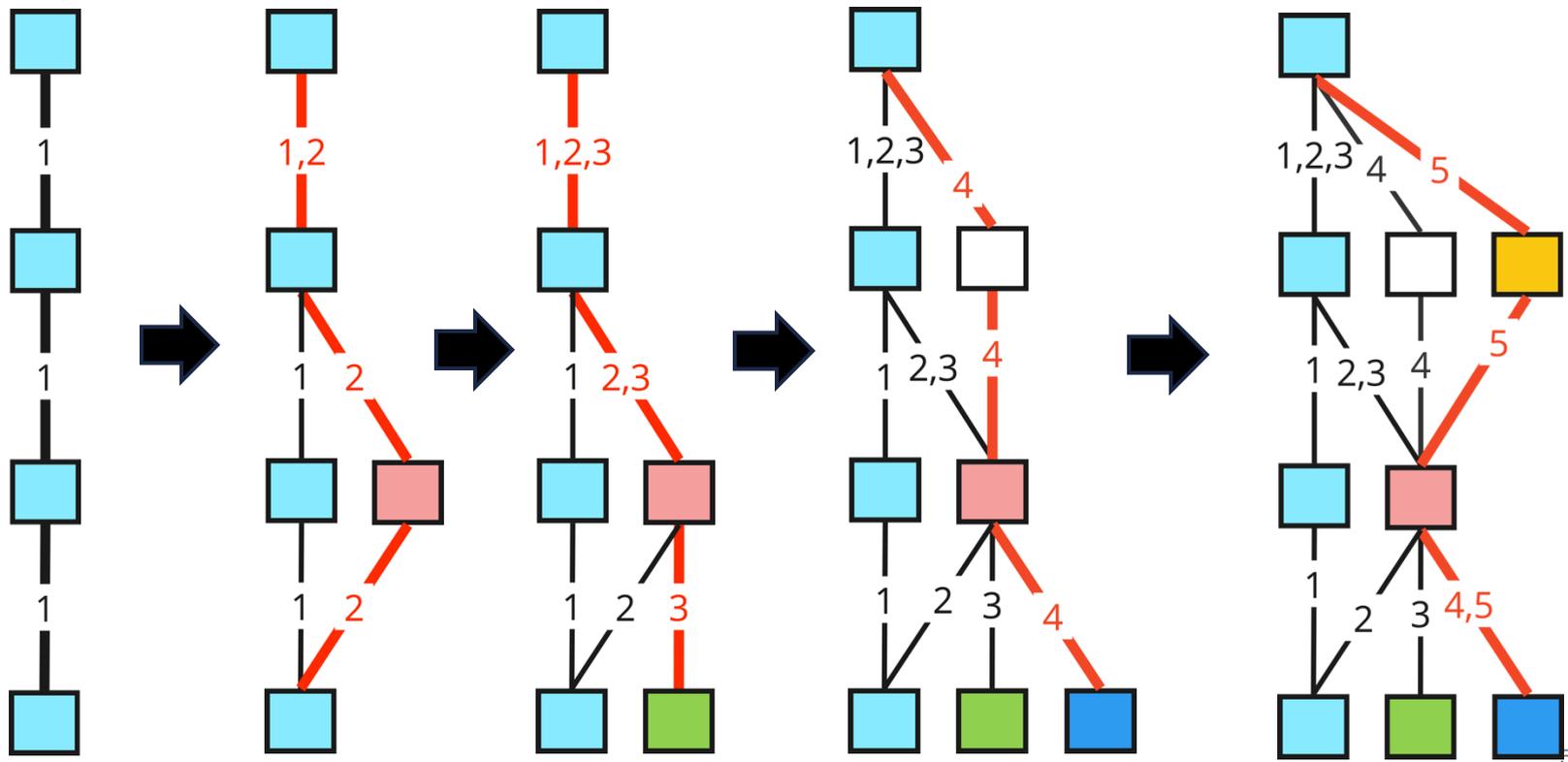
```
In [5]: plt.scatter(petal_length, petal_width, c=clusters)
```

```
Out[5]: <matplotlib.collections.PathCollection at 0x124e294e0>
```

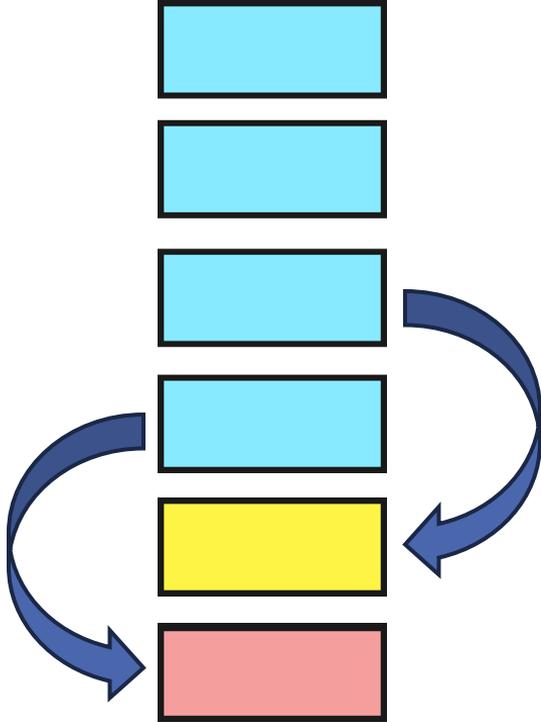


```
In [ ]: |
```

Evolution of ML Pipelines



Hard to Compare Alternatives Using Notebooks



Hard to Compare Alternatives Using Notebooks

≡ kaggle

+ Create

🏠 Home

🏆 Competitions

📁 Datasets

👤 Models

<> Code

💬 Discussions

🎓 Learn

∨ More

🔍 Search

Sign In

Register

Segmentation in PyTorch using convenient tools

Python · [Understanding Clouds from Satellite Images](#)

Notebook Input Output Logs Comments (275)



Competition Notebook

Run

Private Score

[Understanding Clouds from Satellite Ima...](#)

30432.1s - GPU P100

0.63566



Version 37 of 37

Data Visualization

Exploratory Data Analysis

Deep Learning

Computer Vision

General information

In this kernel I work with the data from [Understanding Clouds from Satellite Images](#) competition.

```
Shallow clouds play a huge role in determining the Earth's climate. They're also difficult to understand and to represent in climate models. By classifying different types of cloud or
```

Table of Contents



General information

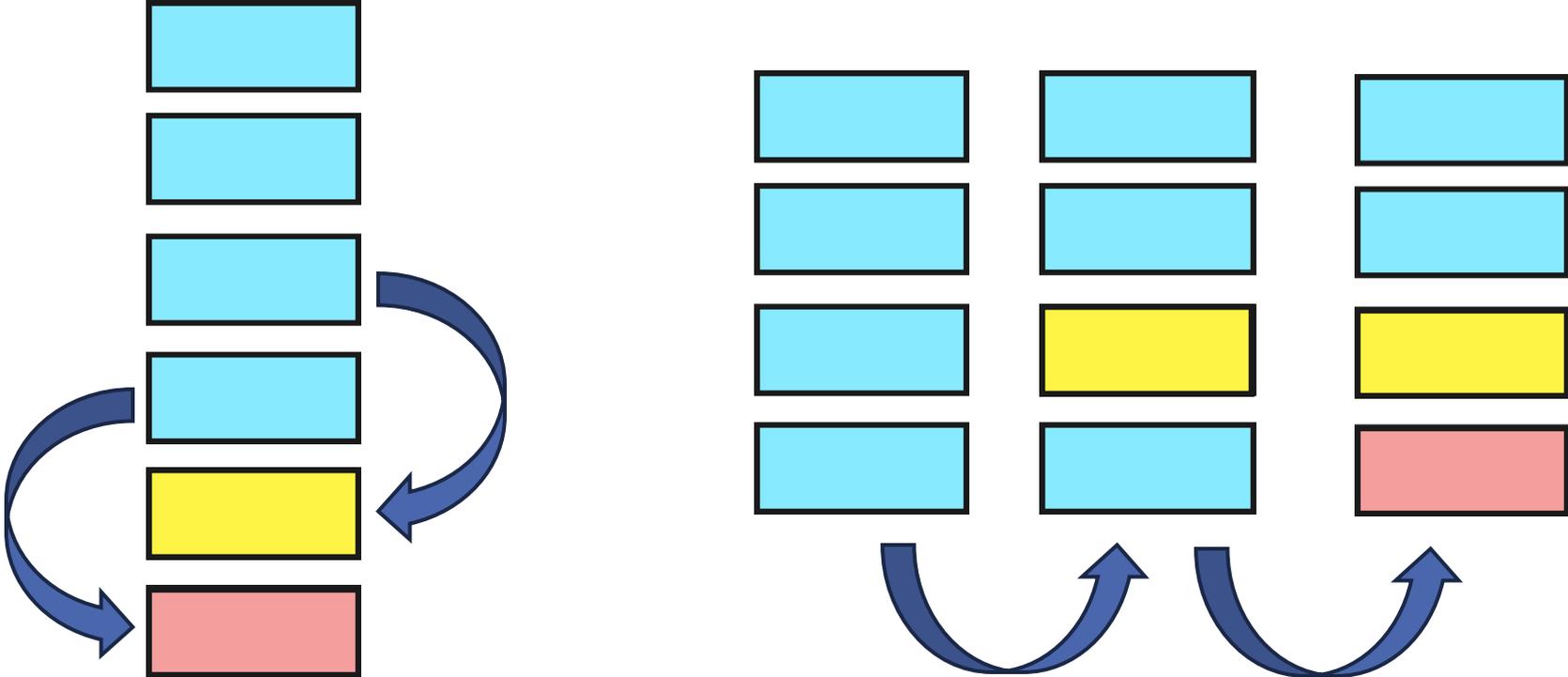
Importing libraries

Helper functions and classes

Data overview

Preparing data for modelling

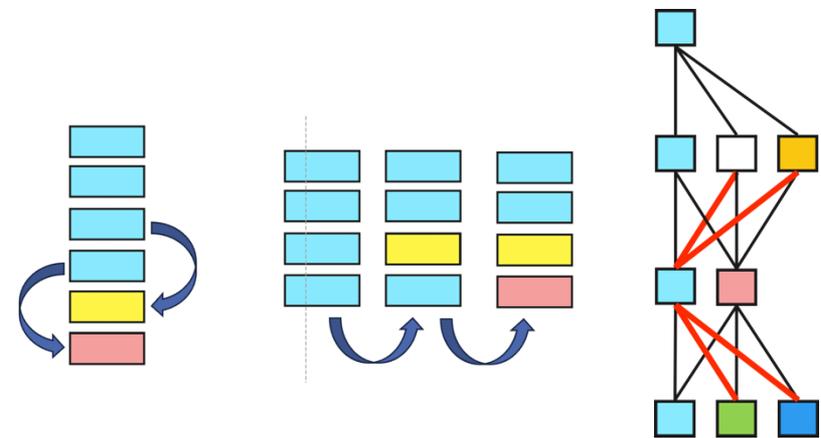
Hard to Compare Alternatives Using Notebooks



Hard to Compare Alternatives Using Notebooks

Version 37	Version 36	Select version to compare
<pre>1 ## General information 2 3 In this kernel I work with the data from Understanding Clouds from Satellite Images competition. ... Expand all 422 unchanged lines (Expand 20 lines) 426 "valid": valid_loader 427 } 428 ----- 429 num_epochs = 19 430 logdir = "./logs/segmentation" 431 432 # model, criterion, optimizer ... Expand all 151 unchanged lines (Expand 20 lines) 584 ----- 585 sub['EncodedPixels'] = encoded_pixels 586 sub.to_csv('submission.csv', columns=['Image_Label', 'EncodedPixels'], index=False)</pre>	<pre>1 ## General information 2 3 In this kernel I work with the data from Understanding Clouds from Satellite Images competition. ... 426 "valid": valid_loader 427 } 428 ----- 429 num_epochs = 21 430 logdir = "./logs/segmentation" 431 432 # model, criterion, optimizer ... 584 ----- 585 sub['EncodedPixels'] = encoded_pixels 586 sub.to_csv('submission.csv', columns=['Image_Label', 'EncodedPixels'], index=False)</pre>	<p>5y ago</p> <p><input checked="" type="checkbox"/> Version 37 Save & Run All • Diff: +1 -1 Ran in 8 hours and 27 minutes</p> <p><input checked="" type="checkbox"/> Version 36 Save & Run All • Diff: +1 -1 Failed after 9 hours</p> <p><input type="checkbox"/> Version 35 Save & Run All • Diff: +53 -28 Ran in 8 hours and 12 minutes</p> <p><input type="checkbox"/> Version 34 Save & Run All • Diff: +1 -1 Ran in 8 hours and 5 minutes</p> <p><input type="checkbox"/> Version 33 Save & Run All • Diff: +1 -1 Cancelled after 7 minutes and 8 seconds</p> <p><input type="checkbox"/> Version 32 Save & Run All • Diff: +1 -1 Ran in 7 hours and 56 minutes</p> <p><input type="checkbox"/> Version 31 Save & Run All • Diff: +2 -2 Ran in 8 hours and 17 minutes</p> <p><input type="checkbox"/> Version 30 Save & Run All • Diff: +3 -3 Ran in 8 hours and 25 minutes</p> <p><input type="checkbox"/> Version 29 Save & Run All • Diff: +2 -2</p>

Observation



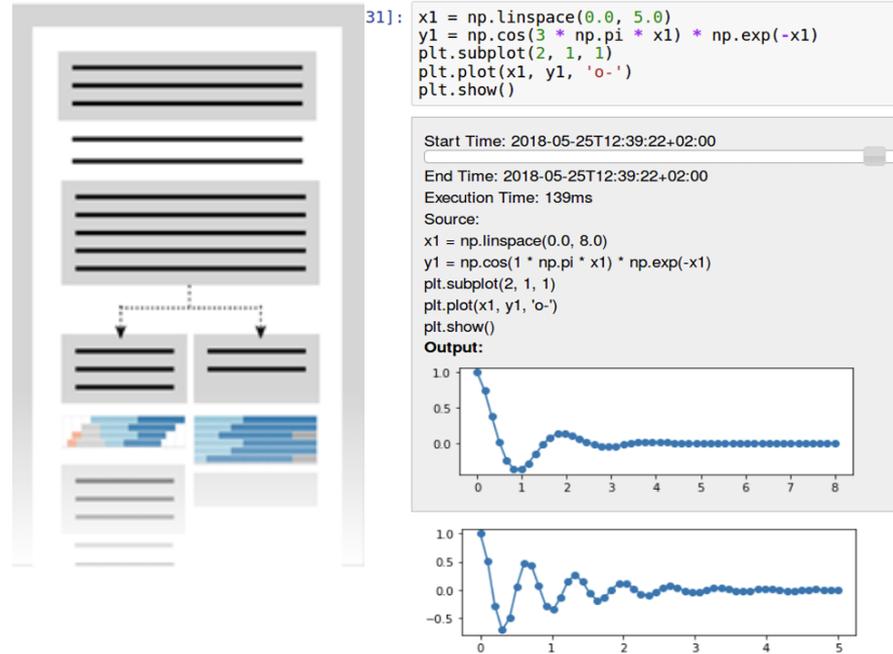
$1 \times 3 \times 2 \times 3 = 18$ possible ML pipelines

Search space can be huge

It might not be feasible to MANUALLY explore all the potential combinations

Prior Work on Supporting Exploration and Versioning in Notebooks

- Integrate a branching mechanism into the notebook [Weinman et al. 2021]
- Track the provenance of cells, enabling the comparison of successive cell versions [Samuel et al. 2018]



HOWEVER

While previous research focused on comparing alternatives for each cell, there is still a need to manually merge alternatives from various ML stages into a new pipeline, and then execute, document, and compare the outcomes.



Long-term Goal

To facilitate the automatic management and exploration of alternatives throughout the exploratory programming process, while preserving the inherent advantages of notebooks.



There remains a lack of systematic understanding of...

- How analysts explore the combination of alternatives across different ML stages?
- If these alternatives are comprehensively analyzed during the ML lifecycle?
- How to further support the exploration after extracting the diffs between versions of notebooks?



Research Questions

Current
practices

Potential of
unexplored ML
pipelines

Research Questions

Current
practices

RQ1: What are the alternatives?
RQ2: How are alternatives explored?

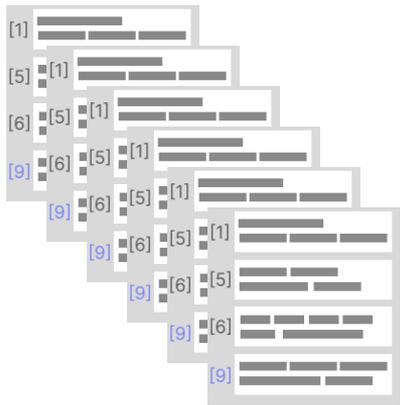
Potential of
unexplored ML
pipelines

Method - RQ1&2 (Current Practices)

MSR + Qualitative analysis



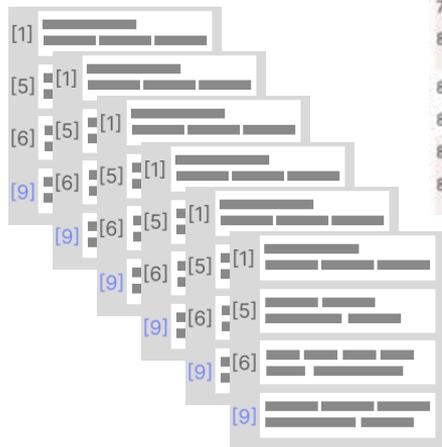
kaggle



52 High-quality Python notebooks, **6** domains, minimum of **5** versions each, **930** versions of ML pipelines, **23385** LOC

Method - RQ1&2 (Current Practices)

Qualitative analysis



```
78 #stem the text
79 text = text.apply(lambda x: " ".join([stemmer.stem(i)
80 for i in re.sub("[^a-zA-Z]", " ", x).split() if i not in
stopwords]).lower())
81
82 #lemmatize the text
83 text = text.apply(lambda x: " ".join([lemmatizer.lemmatize(i)
84 for i in re.sub("[^a-zA-Z]", " ", x).split() if i not in
stopwords]).lower())
```

```
76 #stem the text
77 #text = text.apply(lambda x: " ".join([stemmer.stem(i)
78 #for i in re.sub("[^a-zA-Z]", " ", x).split() if i not in
stopwords]).lower())
79
80 #lemmatize the text
81 #text = text.apply(lambda x: " ".join([lemmatizer.lemmatize(i)
82 #for i in re.sub("[^a-zA-Z]", " ", x).split() if i not in
stopwords]).lower())
```

Version	intention of the code change
1	initial version
2	add comment, finalize code
3	failed
4	failed
5	fix bug
6	create alternative, fix bug
7	no change
8	create alternative
9	failed
10	create alternative



Results - RQ1 (Types of Alternatives)

**Data
Preparation
(DP)**

data cleaning, preprocessing,
and wrangling

**Feature
Engineering
(FE)**

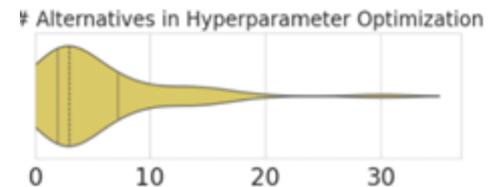
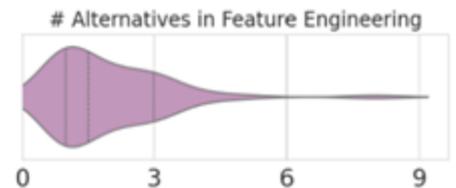
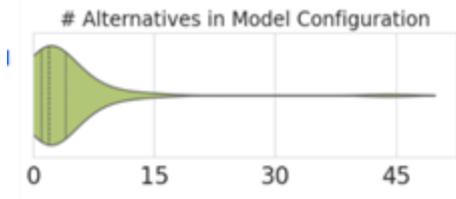
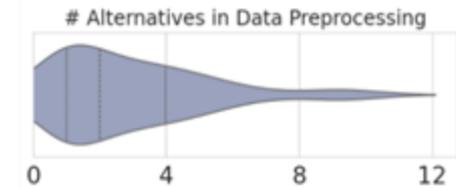
transforming raw data into
relevant features

**Model
Configuration
(MC)**

adjusting the parameters related
to the architecture or individual
components of the model

**Hyperparam
Optimization
(HO)**

adjusting various parameters
that control the learning process

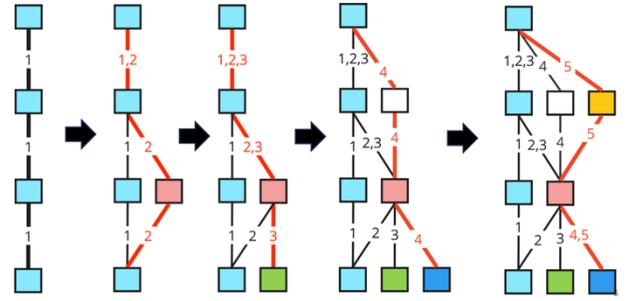


Results - RQ2

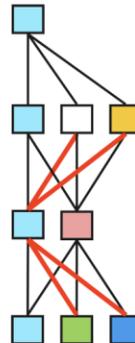
(How are alternatives explored?)

An iterative fashion

The median of our selected notebooks only represents 1.8% of all possible combinations.



Comparing Alternatives can be Tedious



$1 \times 3 \times 2 \times 3 = 18$ possible ML pipelines

The prior example only explored 5 pipelines

Search space can be huge

#ML pipelines is exponential

It might not be feasible to explore all the potential combinations

Research Questions

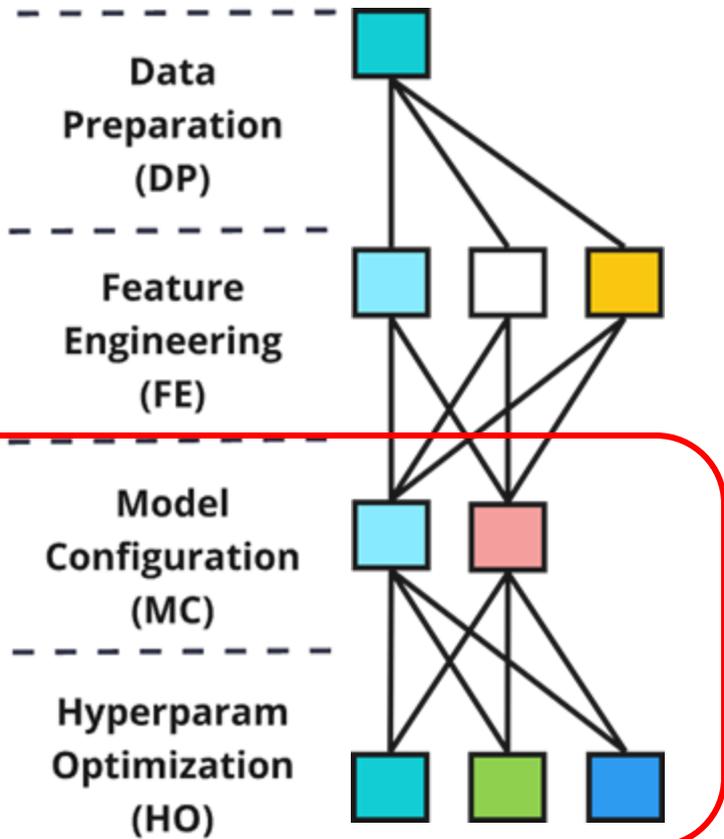
Current practices

RQ1: What are the alternatives?
RQ2: How are alternatives explored?

Potential of unexplored ML pipelines

RQ3: Evaluating unexplored ML pipeline
RQ4:
RQ5:

RQ4: Potential and capability of AutoML



replaced by



Research Questions

Current practices

RQ1: What are the alternatives?

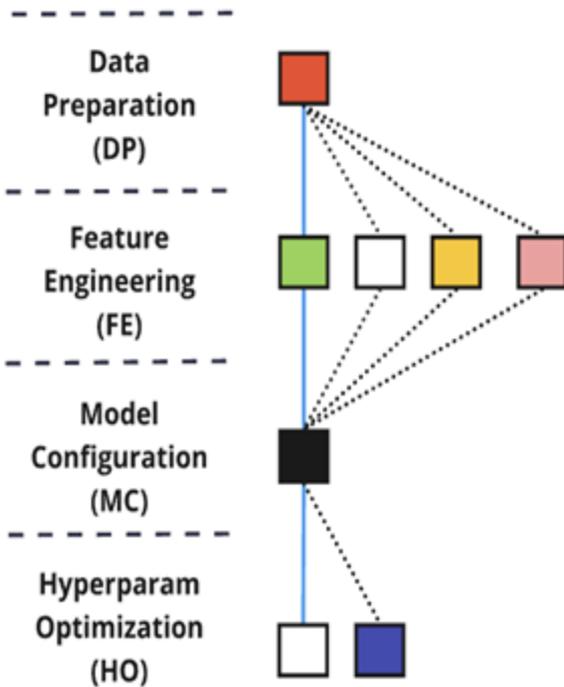
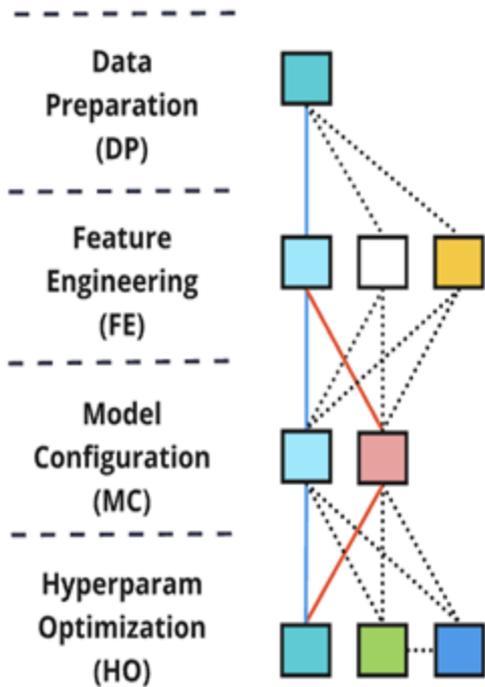
RQ2: How are alternatives explored?

Potential of unexplored ML pipelines

RQ3: Evaluating unexplored ML pipeline

RQ4: Potential and capability of AutoML

RQ5: Feasibility of Combining Alternatives from Different Analysts



RQ5: Will a combination of alternatives from different data scientists outperform the original notebook result? If yes, to what extent?

Research Questions

Current practices

RQ1: What are the alternatives
RQ2: How are alternatives explored?

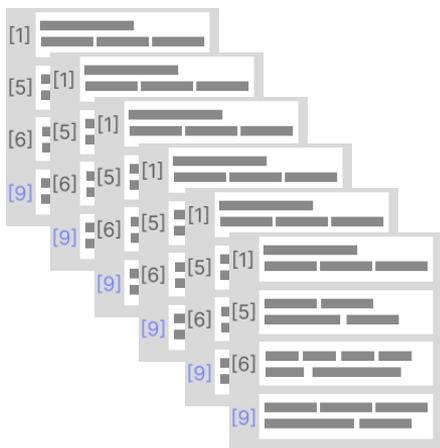
Potential of unexplored ML pipelines

RQ3: Evaluating unexplored ML pipeline
RQ4: Potential and capability of AutoML
RQ5: Crowdsourcing alternatives

Method to RQ3-5 (Unexplored ML Pipelines)

Quantitative analysis

kaggle



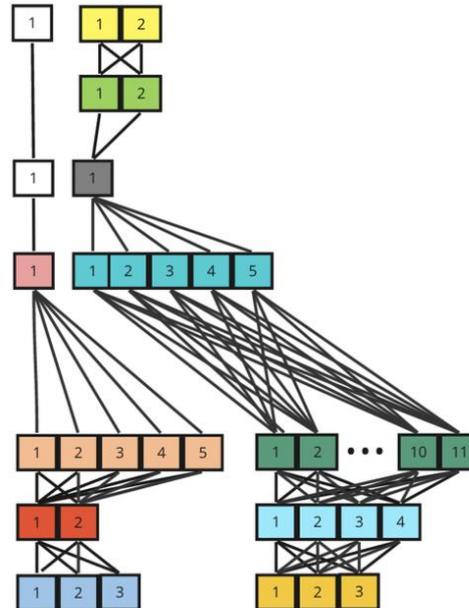
20 notebooks

11787 ML pipelines in total

Method - RQ3-5

- If the search space is too big, we conduct experiments by randomly sampling a subset of pipelines.

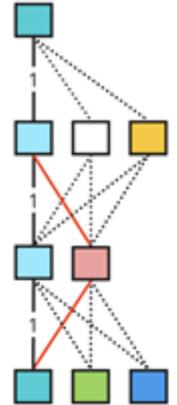
NB5 has a total of 44,236,800 pipelines with a total approximated running time of 70,707,610 hrs



Results - RQ3

Potential of Unexplored Pipelines

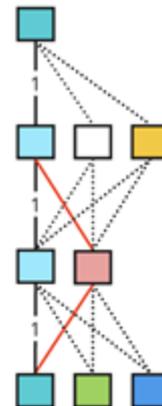
- 19/20 NBs contain unexplored pipelines



Results - RQ3

Potential of Unexplored Pipelines

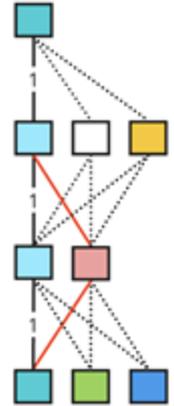
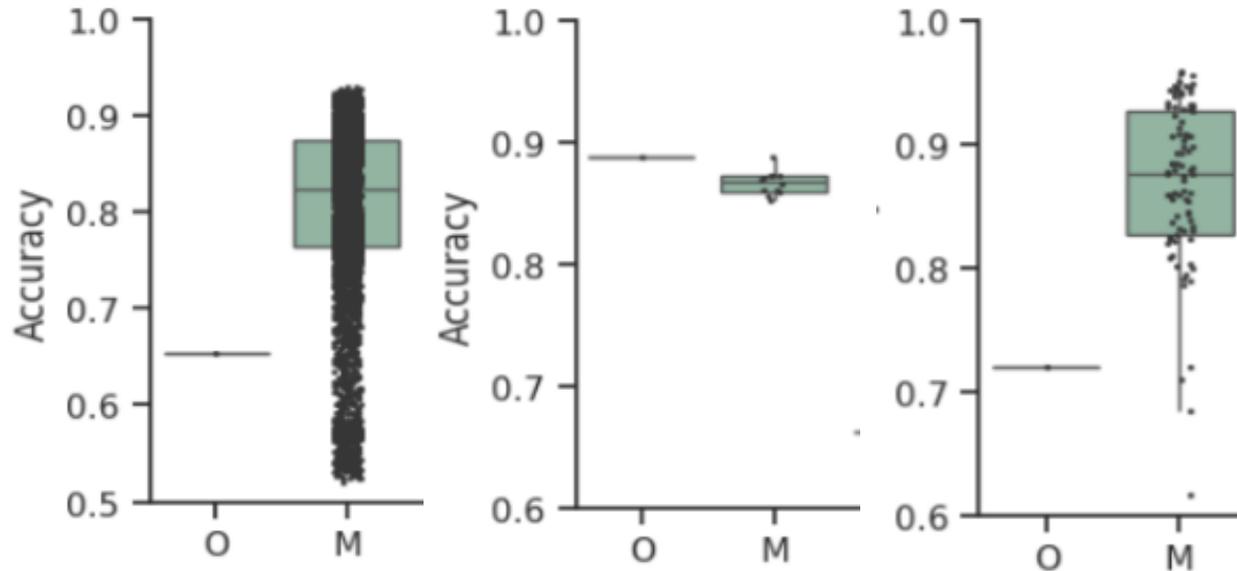
- 19/20 NBs contain unexplored pipelines
- 16 NBs has performance increase
(Avg 13.57%)
- 3 NBs have no performance increase



Results - RQ3

Potential of Unexplored Pipelines

- 19/20 NBs contain unexplored pipelines



Research Questions

Current practices

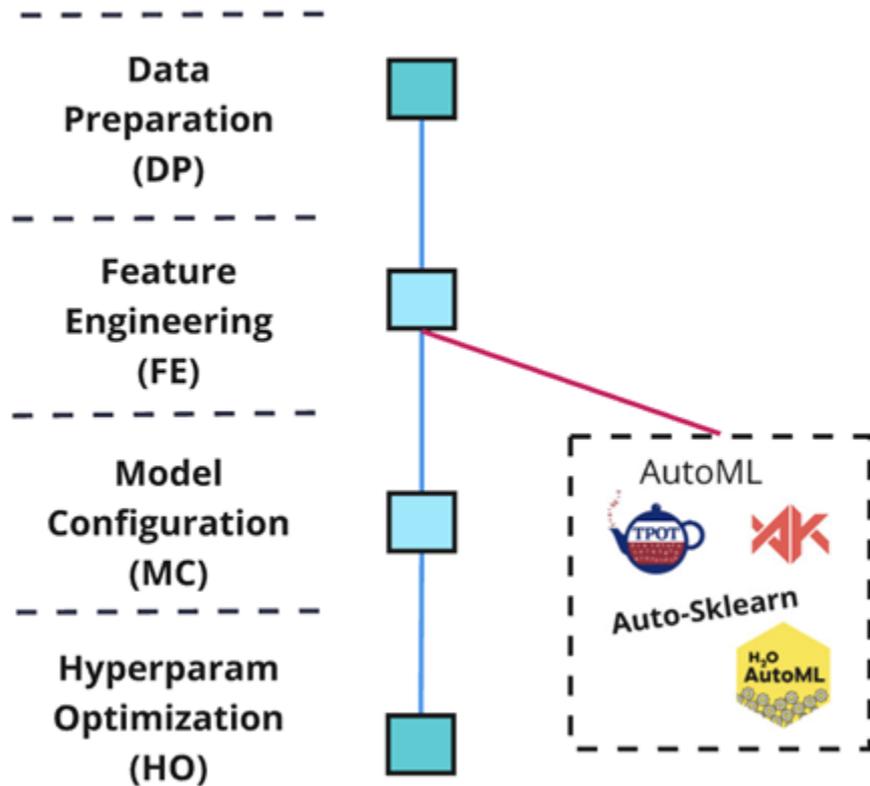
RQ1: What are the alternatives
RQ2: How are alternatives explored?

Potential of unexplored ML pipelines

RQ3: Evaluating unexplored ML pipeline
RQ4: Potential and capability of AutoML
RQ5: Crowdsourcing alternatives

Results – RQ4

Evaluating AutoML in Exploratory Programming



Results – RQ4

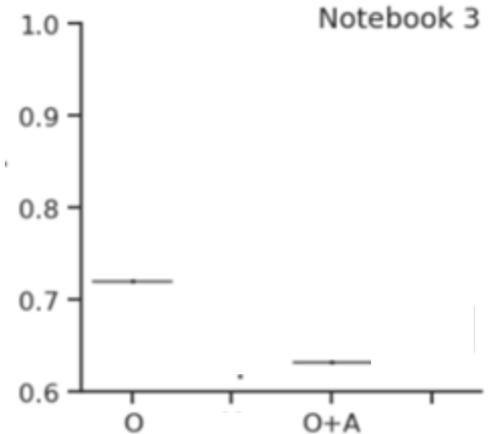
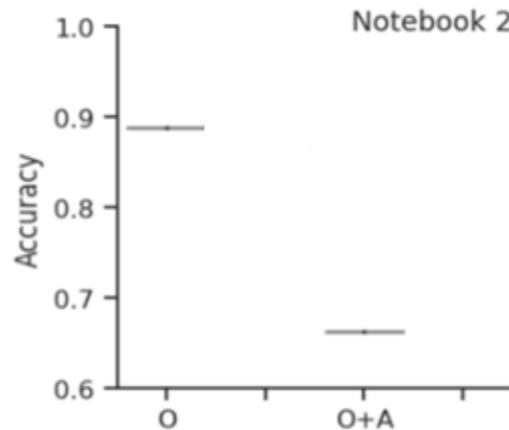
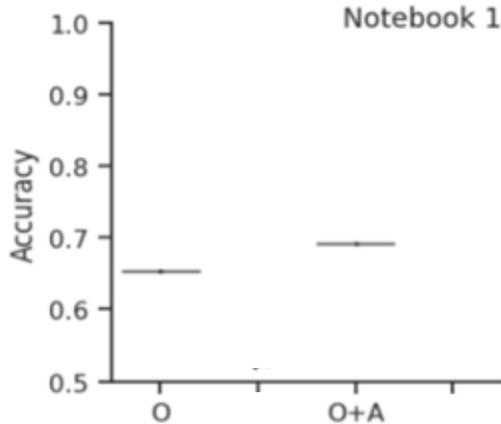
Original pipeline VS Original pipeline + AutoML



- 3/20 NBs show an average performance **increase** of 8.72%.



- 17/20 NBs show an average performance **decrease** of 16.59%.

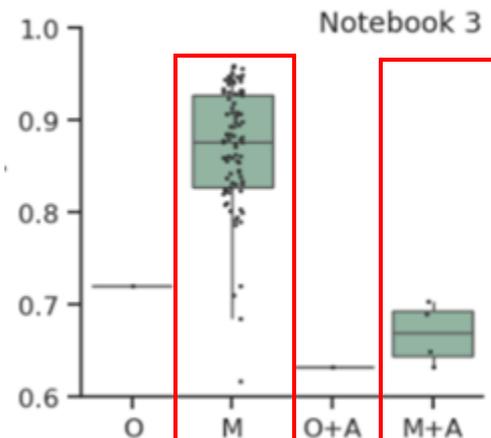
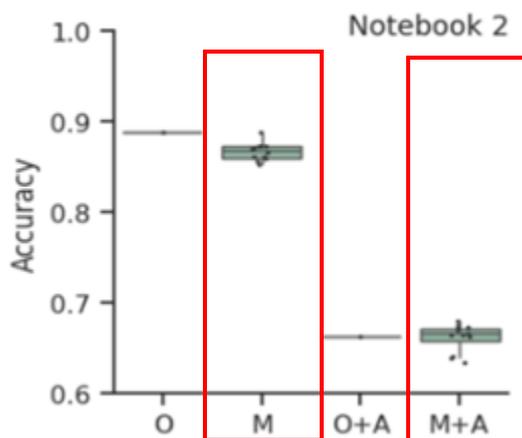
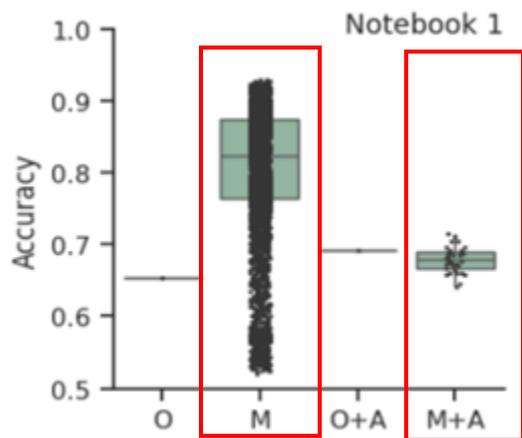


Results of RQ4 (Cont.):

Merged pipeline VS Merged pipeline + AutoML



- 1/20 NBs have 8.14% performance increase due to a better set of hyperparameters
- 19/20 NBs have avg 17.41% performance decrease



Research Questions

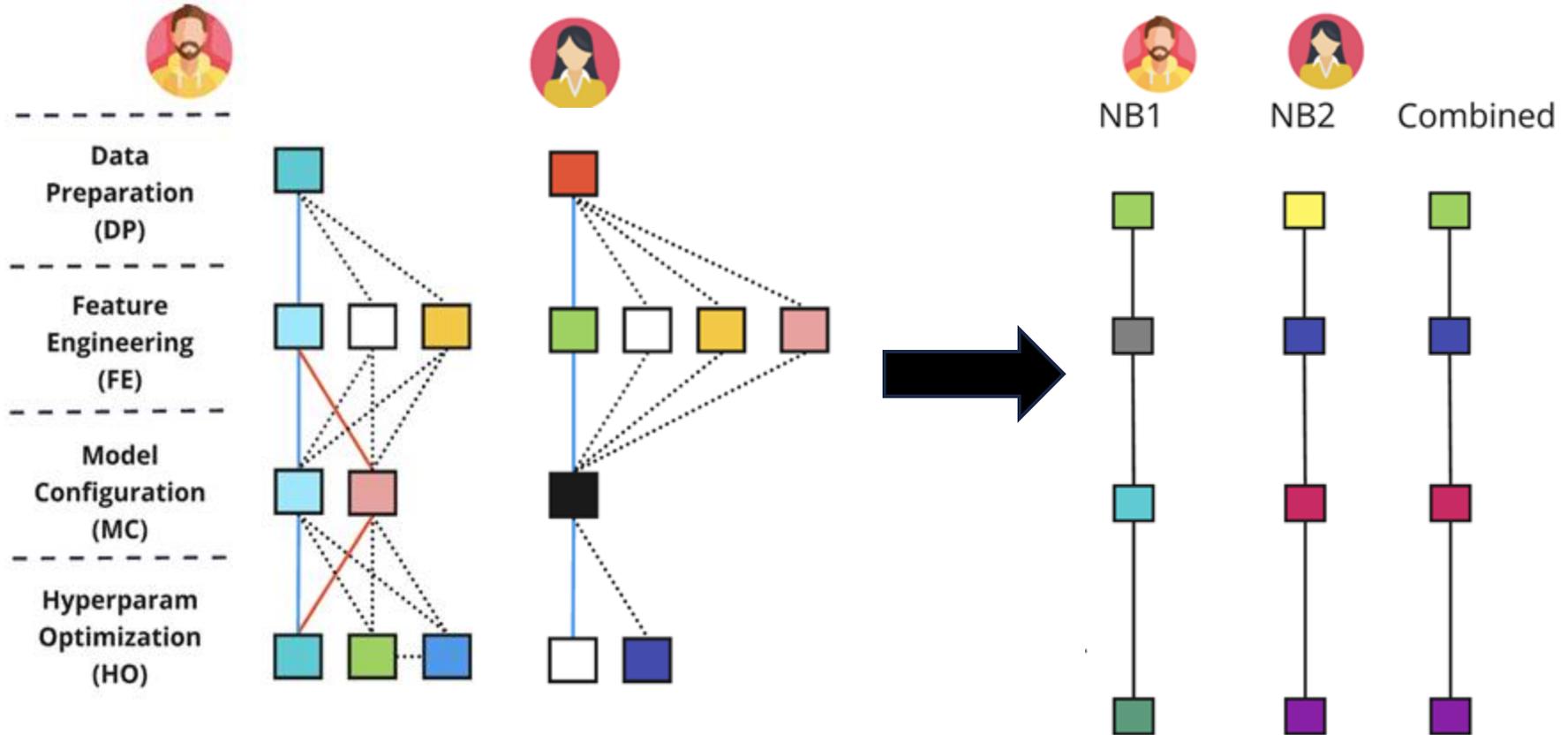
Current practices

RQ1: What are the alternatives
RQ2: How are alternatives explored?

Potential of unexplored ML pipelines

RQ3: Evaluating the unexplored ML pipeline
RQ4: Potential and capability of AutoML
RQ5: Crowdsourcing alternatives

RQ5: Crowdsourcing Alternatives



Result - RQ5: Crowdsourcing Alternatives



3/10 NBs - avg 21.61%



4/10 NBs - avg 11.47%

3/10 NBs show
operational
errors

Finding: Despite the potential for improvement demonstrated in some notebooks, a **large amount of manual effort** is usually required to remove these inconsistencies for a successful integration.

Research Questions

RECAP

Current practices

RQ1: What are the alternatives

RQ2: How are alternatives explored?

Potential of unexplored ML pipelines

RQ3: Evaluating the unexplored ML pipeline

RQ4: Potential and capability of AutoML

RQ5: Crowdsourcing alternatives

Tooling Opportunities & Research Directions

- More efficient ways to extracting and managing alternatives from large number of notebooks

Version	intention of the code change
1	initial version
2	add comment, finalize code
3	failed
4	failed
5	fix bug
6	create alternative, fix bug
7	no change
8	create alternative
9	failed
10	create alternative

```
78 #stem the text
79 text = text.apply(lambda x: " ".join([stemmer.stem(i)
80 for i in re.sub("[^a-zA-Z]", " ", x).split() if i not in
stopwords]).lower())
81
82 #lem
83 text = text.apply(lambda x: " ".join([lemmatizer.lemmatize(i)
84 for i in re.sub("[^a-zA-Z]", " ", x).split() if i not in
stopwords]).lower())
```



Tooling Opportunities & Research Directions

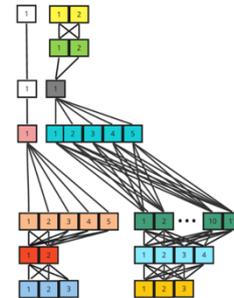
- More efficient ways to extracting and managing alternatives from large number of notebooks
- **More automated ways combine and execute merged pipelines is needed**

SPL?

Method - RQ3-5

- If the search space is extensive, we conduct experiments by randomly sampling a subset of pipelines.

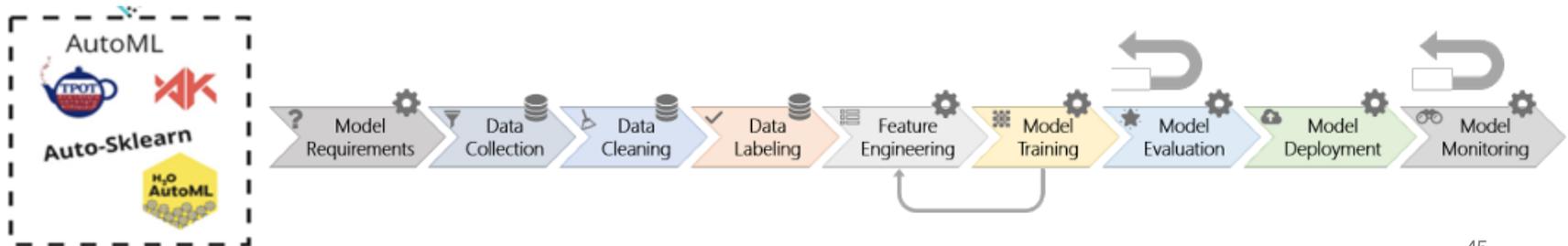
NB5 has a total of 44,236,800 pipelines with a total approximated running time of 70,707,610 hrs



29

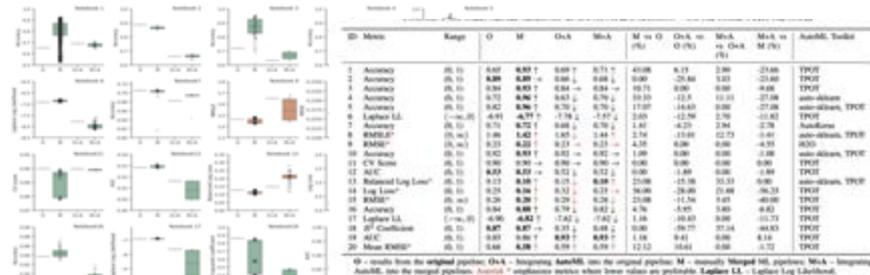
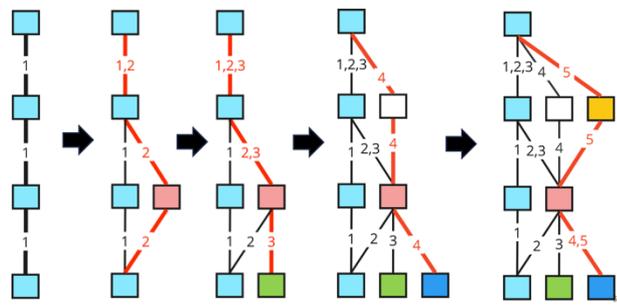
Tooling Opportunities & Research Directions

- More efficient ways to extracting and managing alternatives from large number of notebooks
- More efficient ways combine and execute merged pipelines is needed
- **Better usability of AutoML tools during exploratory data analysis [Alamin et al. 2022]**



Can We Do Better with What We Have Done?

Unveiling the Potential of ML Pipeline in Notebooks



Research Questions

Current practices
 RQ1: What are the alternatives
 RQ2: How are alternatives explored?

Potential of unexplored ML pipelines
 RQ3: Evaluating the unexplored ML pipeline
 RQ4: Potential and capability of AutoML
 RQ5: Crowdsourcing alternatives

Tooling Opportunities & Research Directions

- More efficient ways to extracting and managing alternatives from large number of notebooks
- More efficient ways combine and execute merged pipelines is needed
- Better usability of AutoML tools during exploratory data analysis [Alamin et al. 2022]